

Kenneth Faurillo 201812633

## ME 8: Containerization and Docker

### Activity 6

#### Inspecting bridge network

```
Windows PowerShell
PS Y:\wsl\197\ME8_Containerization_and_Docker\5-volumes> docker network ls
NETWORK ID      NAME      DRIVER      SCOPE
ba534af57e54    bridge   bridge      local
ba879c76c74e    host     host        local
3f14cbcea4f2    none     null        local
PS Y:\wsl\197\ME8_Containerization_and_Docker\5-volumes> docker network inspect bridge
[
  {
    "Name": "bridge",
    "Id": "ba534af57e540a30d2bf3c0340eb3812583916fbfe59fafe6a66e2dfcc8b7ee5",
    "Created": "2021-05-26T08:18:57.5224837Z",
    "Scope": "local",
    "Driver": "bridge",
    "EnableIPv6": false,
    "IPAM": {
      "Driver": "default",
      "Options": null,
      "Config": [
        {
          "Subnet": "172.17.0.0/16",
          "Gateway": "172.17.0.1"
        }
      ]
    },
    "Internal": false,
    "Attachable": false,
    "Ingress": false,
    "ConfigFrom": {
      "Network": ""
    },
    "ConfigOnly": false,
    "Containers": {},
    "Options": {
      "com.docker.network.bridge.default_bridge": "true",
      "com.docker.network.bridge.enable_icc": "true",
      "com.docker.network.bridge.enable_ip_masquerade": "true",
      "com.docker.network.bridge.host_binding_ipv4": "0.0.0.0",
      "com.docker.network.bridge.name": "docker0",
      "com.docker.network.driver.mtu": "1500"
    }
  }
]
```

#### Starting a ping container then inspecting the bridge network

```
Windows PowerShell
PS Y:\wsl\197\ME8_Containerization_and_Docker\5-volumes> docker run --rm -d --name dummy kennethfau/ping
ee9f7eba08086397e7398a13952b189a59c9f082f8f8e3eaff252486f71cee0c
PS Y:\wsl\197\ME8_Containerization_and_Docker\5-volumes> docker network inspect bridge
[
  {
    "Name": "bridge",
    "Id": "ba534af57e540a30d2bf3c0340eb3812583916fbfe59fafe6a66e2dfcc8b7ee5",
    "Created": "2021-05-26T08:18:57.5224837Z",
    "Scope": "local",
    "Driver": "bridge",
    "EnableIPv6": false,
    "IPAM": {
      "Driver": "default",
      "Options": null,
      "Config": [
        {
          "Subnet": "172.17.0.0/16",
          "Gateway": "172.17.0.1"
        }
      ]
    },
    "Internal": false,
    "Attachable": false,
    "Ingress": false,
    "ConfigFrom": {
      "Network": ""
    },
    "ConfigOnly": false,
    "Containers": {
      "ee9f7eba08086397e7398a13952b189a59c9f082f8f8e3eaff252486f71cee0c": {
        "Name": "dummy",
        "EndpointID": "05e6dc66a18e4a95e95df40b488761319af537953f844dd9664d7d181fa04108",
        "MacAddress": "02:42:ac:11:00:02",
        "IPv4Address": "172.17.0.2/16",
        "IPv6Address": ""
      }
    },
    "Options": {
      "com.docker.network.bridge.default_bridge": "true",
      "com.docker.network.bridge.enable_icc": "true",
    }
  }
]
```

## Creating a new ping container to ping the first one then inspecting its logs

```
Windows PowerShell
PS Y:\wsl\197\ME8_Containerization_and_Docker\5-volumes> docker run --rm -d -e PING_TARGET=172.17.0.2 --name pinger kennethfau/ping
713474524addef0f78743321258ddcbfb9602d9dada27fe785df1224c15664c8
PS Y:\wsl\197\ME8_Containerization_and_Docker\5-volumes> docker logs pinger
PING 172.17.0.2 (172.17.0.2) 56(84) bytes of data.
64 bytes from 172.17.0.2: icmp_seq=1 ttl=64 time=0.091 ms
64 bytes from 172.17.0.2: icmp_seq=2 ttl=64 time=0.065 ms
64 bytes from 172.17.0.2: icmp_seq=3 ttl=64 time=0.087 ms
64 bytes from 172.17.0.2: icmp_seq=4 ttl=64 time=0.086 ms
64 bytes from 172.17.0.2: icmp_seq=5 ttl=64 time=0.090 ms
64 bytes from 172.17.0.2: icmp_seq=6 ttl=64 time=0.045 ms
PS Y:\wsl\197\ME8_Containerization_and_Docker\5-volumes>
```

## Creating a custom network and pinging the dummy ping container

```
Windows PowerShell
PS Y:\wsl\197\ME8_Containerization_and_Docker\5-volumes> docker network ls
NETWORK ID          NAME                DRIVER              SCOPE
ba534af57e54        bridge             bridge             local
ba879c76c74e        host               host               local
3f14cbcea4f2        none              null               local
a9cdb61caf75        skynet            bridge             local
PS Y:\wsl\197\ME8_Containerization_and_Docker\5-volumes> docker run --rm -d --network skynet --name dummy kennethfau/ping
6ccf92576360b7acc59c480580554e1f865a9b7a581744be6e4ddfd1ed416f6e
PS Y:\wsl\197\ME8_Containerization_and_Docker\5-volumes> docker run --rm -d --network skynet -e PING_TARGET=dummy --name pinger delner/ping:1.0
Unable to find image 'delner/ping:1.0' locally
docker: Error response from daemon: pull access denied for delner/ping, repository does not exist or may require 'docker login': denied: requested access to
the resource is denied.
See 'docker run --help'.
PS Y:\wsl\197\ME8_Containerization_and_Docker\5-volumes> docker run --rm -d --network skynet -e PING_TARGET=dummy --name pinger kennethfau/ping
0652106ff531d221c25756dfa3def528128ab0a6e8c8a7f7159b8be30a571664
PS Y:\wsl\197\ME8_Containerization_and_Docker\5-volumes> docker logs pinger
PING dummy (172.18.0.2) 56(84) bytes of data.
64 bytes from dummy.skynet (172.18.0.2): icmp_seq=1 ttl=64 time=0.100 ms
64 bytes from dummy.skynet (172.18.0.2): icmp_seq=2 ttl=64 time=0.118 ms
64 bytes from dummy.skynet (172.18.0.2): icmp_seq=3 ttl=64 time=0.050 ms
64 bytes from dummy.skynet (172.18.0.2): icmp_seq=4 ttl=64 time=0.049 ms
PS Y:\wsl\197\ME8_Containerization_and_Docker\5-volumes>
```

## Starting widgetdb and gadgetdb postgres databases

```
Windows PowerShell
PS C:\Users\Kenneth> docker run --rm -d --name gadgetdb --network skynet -p 5432 -e POSTGRES_PASSWORD=password postgres
c235a3aeadd4aefec77841fbeb475ef9a06927b325b7fe708c49fe52ddc2ab2d
PS C:\Users\Kenneth> docker ps
CONTAINER ID   IMAGE      COMMAND                  CREATED          STATUS          PORTS                               NAMES
c235a3aeadd4   postgres  "docker-entrypoint.s..." Less than a second ago Up 1 second     0.0.0.0:59603->5432/tcp             gadgetdb
PS C:\Users\Kenneth> docker run --rm -d --name widgetdb --network skynet -p 5432 -e POSTGRES_PASSWORD=password postgres
0bc41a14f68f7a5e3b78bb2d73f4c94609317f676c784b5fe96c25c4d298cc7a
PS C:\Users\Kenneth> docker ps
CONTAINER ID   IMAGE      COMMAND                  CREATED          STATUS          PORTS                               NAMES
0bc41a14f68f   postgres  "docker-entrypoint.s..." 3 seconds ago    Up 4 seconds    0.0.0.0:59675->5432/tcp             widgetdb
c235a3aeadd4   postgres  "docker-entrypoint.s..." 23 seconds ago   Up 24 seconds   0.0.0.0:59603->5432/tcp             gadgetdb
PS C:\Users\Kenneth> docker exec -it widgetdb /bin/bash
root@0bc41a14f68f:/#
```

## Connecting to the databases

```
Windows PowerShell
PS C:\Users\Kenneth> docker exec -it widgetdb /bin/bash
root@0bc41a14f68f:/# psql -U postgres
psql (13.3 (Debian 13.3-1.pgdg100+1))
Type "help" for help.

postgres=# \q
root@0bc41a14f68f:/# psql -U postgres -h gadgetdb
Password for user postgres:
psql (13.3 (Debian 13.3-1.pgdg100+1))
Type "help" for help.

postgres=# \q
root@0bc41a14f68f:/# exit
exit
PS C:\Users\Kenneth> docker stop widgetdb
widgetdb
PS C:\Users\Kenneth> docker stop gadgetdb
gadgetdb
PS C:\Users\Kenneth> |
```

## Binding ports to the host

```
Windows PowerShell
PS C:\Users\Kenneth> docker run --rm -d --name widgetdb --network skynet -p 5432:5432 -e POSTGRES_PASSWORD=password postgres
e0181a519237d9656af788e92b287215eb03f2b335c01a456eb7691d99c0cfff4
PS C:\Users\Kenneth> docker ps
CONTAINER ID   IMAGE      COMMAND                  CREATED        STATUS        PORTS                               NAMES
e0181a519237   postgres   "docker-entrypoint.s..." About a minute Up About a minute 0.0.0.0:5432->5432/tcp, :::5432->5432/tcp widgetdb
PS C:\Users\Kenneth> docker stop widgetdb
widgetdb
PS C:\Users\Kenneth> |
```